process additional user requests. But the server load itself is dependant on the number of messages (N) being passed between the server 16 and the client 14. As a result, the service time (S) can be dependent on the number of messages (N). Similarly, the time spent in communications (C) can also depend on the number of messages (N) broadcast over the network 12 because network bandwidth can limit the amount of traffic carried by the network. Accordingly, when the number of messages (N) increases above a certain threshold, the service time (S) may increase because of the increased load on the server and the time spent in network ~~communication~~ communications (C) may increase because of network congestion. Therefore, there is an indirect relationship between the service time (S) and the time spent in communications (C) because of their mutual dependence on the number of messages (N).

On page 16, please replace the paragraph beginning on line 1 with the following paragraph:

Second, the time spent in ~~communications(C)~~communications (C) is generally independent of the time spent in fault tolerance (FT). However, the time spent in fault tolerance (FT), which includes the fault tolerance times on the client 14 and the server 16, can also depend at any given instant on the load on the server and the client. Therefore, the time spent in fault tolerance (FT) can depend on the number of messages (N) being passed between the client 14 and the server 16. As a result, there is an indirect relationship between the time spent in ~~communications(C)~~communications (C) and the time spent in fault tolerance (FT) because of their mutual dependence on the number of messages (N).

On page 17, please replace the paragraph beginning on line 16 with the following paragraph:

As a first block 32, the switching algorithm 30 obtains values of measured wait times W and calculates a value of the mean wait time ~~(m(W))for~~(m(W)) for an application. For example, the client 14 and server 16 in the distributed system of FIG. 1 can use timestamps for actions associated with user interface events of the distributed application 24 in order to measure wait times (W). Specifically, an HTML based client 14 can intercept all HTTP "GET" and "POST" requests from a web browser type client application 20 to the server application 22. When a "GET" or "POST" request is issued, the client 14 takes a first timestamp. When the "GET" or "POST" request returns and the reply generated by the server application is displayed using the browser, a second timestamp is taken by the client 14. The measured wait time (W) in this case is the difference between the second and first timestamps. The mean wait time (m(W) then is calculated from a plurality of measured values for wait times (W) using known statistical methods.